AD-A234 422

REPORT DOCUMENTATION PAGE

Form Approved OMB No 0704-0188

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE October 1990	3. REPORT TYPE AND DATES COVERED memorandum

4. TITLE AND SUBTITLE

Line Kinematics for Whole-Arm Manipulation

5. FUNDING NUMBERS

N00014-86-K-0685
N00014-85-K-0124
75-2608

6. AUTHOR(S)

Brian Eberman and David L. Brock

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Artificial Intelligence Laboratory
545 Technology Square
Cambridge, Massachusetts 02139

8. PERFORMING ORGANIZATION REPORT NUMBER

AIM 1255

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)

Office of Naval Research
Information Systems
Arlington, Virginia 22217

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

None

12a. DISTRIBUTION/AVAILABILITY STATEMENT

Distribution of this document is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(Maximum 200 words)*

Abstract: A Whole-Arm Manipulator uses every surface to both sense and interact with the environment. To facilitate the analysis and control of a Whole-Arm Manipulator, line geometry is used to describe the location and trajectory of the links. Applications of line kinematics are described and implemented on the MIT Whole-Arm Manipulator (WAM-1).

14. SUBJECT TERMS (key words)

kinematics, lines, whole-arm, manipulation

15. NUMBER OF PAGES 13

16. PRICE CODE $3.00

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED

MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I.Memo No. 1255 January, 1991

Line Kinematics for Whole-Arm Manipulation

Brian Eberman
David L. Brock

Abstract: A Whole-Arm Manipulator uses every surface to both sense and interact with the environment. To facilitate the analysis and control of a Whole-Arm Manipulator, line geometry is used to describe the location and trajectory of the links. Applications of line kinematics are described and implemented on the MIT Whole-Arm Manipulator (WAM-1).

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for this research has been provided in part by Sandia National Laboratories under contract 75-2608, in part by the Office of Naval Research University Research Initiative Program under Office of Naval Research contract N00014-86-K-0685 and in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-85-K-0124.

1 Introduction

Whole-arm Manipulation is the description and control of every active surface of a robot [1, 2, 3]. In the real, unstructured environment contacts at locations other than the end-effector are both inevitable and useful. Pushing and shoving [4] as well as striking, cradling and inter-link grasping are examples of such operations. People frequently use their arms to hold a bundle of logs, grasp a barrel, snag a loop, cushion a blow or push off from a constraint. To get the maximum utility from expensive manipulators in critical and unstructured situations, we should not preclude such operations from the arm's repertoire. With this perspective, the distinction between fingers, arms and multiple cooperating robots begins to blur.

The description and control of these types of motions and effectors provides an opportunity to reexamine traditional manipulator motion control. The choice of a kinematic system should be guided by the utility of the system for describing the tasks of interest. For end-point manipulation where the concern is the orienting and positioning of an object that is affixed to the robot at a given point a Cartesian frame of reference is clearly of high utility. However, consider a manipulation problem where the point of interaction is not specifically described, and where a range of possible interaction locations is allowed. In this case, specifying a motion for a fixed point on the manipulator may be overly constraining and may not fit the kinematics and *motion constraints* of the task. In particular, for pushing objects with the links of a manipulator, or for examining the workspace of a robot (searching) a different kinematic approach called **line kinematics** has proven useful. In this approach, one or more links of the manipulator are controlled to lie along a desired line.

In this paper we will describe line kinematics and its application to robotic manipulation. We introduce the idea of controlling the trajectory of a line that passes through the link of a manipulator. We will review the basic concepts of line theory and present the line Jacobian, line-workspace, and stiffness of a line. We will also apply these concepts to the WAM-1 including the calculation of both forward and inverse line kinematics and generation of bounded deviation line trajectories.

It should be noted that the line segment formed by a manipulator's link will have some friction along its length. This implies that this "line" can exert five forcesf and not the usual four associated with a frictionless line. However, in all cases the four forces associated with a line will be the most reliable. Additionally, we will be focusing on kinematics, or positioning, for which a line description is sufficient.

This work was demonstrated on a 4-degree-of-freedom manipulator that has been designed and built at the Massachusetts Institute of Technology's Artificial Intelligence Lab (the WAM-1) as shown in figure 1. A description of this demonstration is provided.

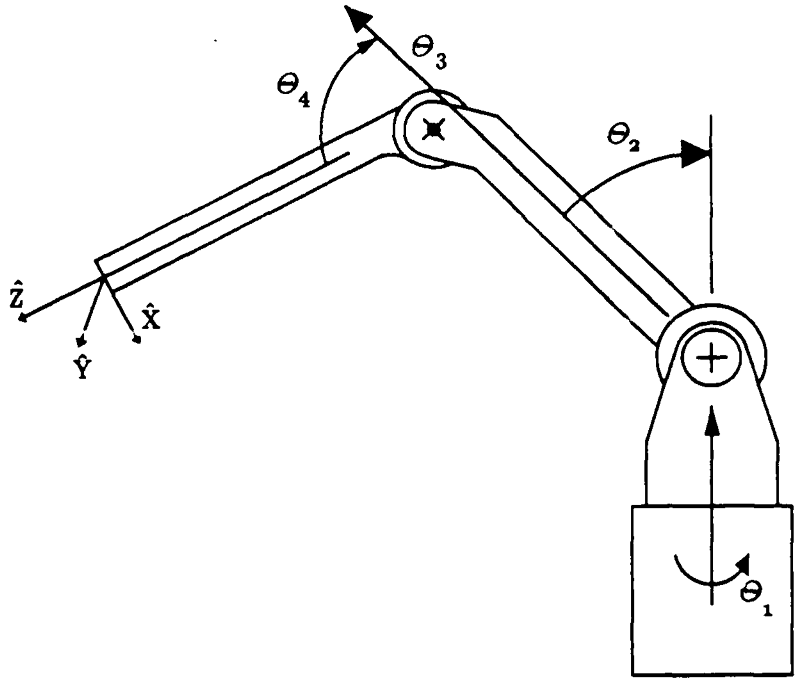

Figure 1: MIT Whole-Arm Manipulator.

2 Line Geometry

2.1 Line Representation

The representation of lines as fundamental spatial elements was originally conceived by Plücker in 1846; however the following notation is due to Grassmann in 1844 and the quadratic relation to Cayley in 1859, [5] and [6]. The result is what has become known as the Plücker coordinates for a line in space. Let $x_1, \ldots, x_4$ and $y_1, \ldots y_4$ be the homogeneous coordinates of two points on a line. Then the sextuple $c = (c_{41}, c_{42}, c_{43}, c_{23}, c_{31}, c_{12})$ where

$$c_{ij} = x_i y_j - x_j y_i$$

defines the six homogeneous Plücker coordinates of a line in space. Lines in space are in one-to-one correspondence with the equivalence classes of Plücker coordinates under the relation $c \sim c'$ if $c = \lambda c'$ for some $\lambda \in \Re$. In our analysis we will use a normalization of the Plücker coordinates, to a constant ρ, to represent lines. Let L_d denote the space of directed lines,

$$L_d = \{l | l = (\hat{l}, \bar{l}) | \hat{l}, \bar{l} \in \Re^3, |\hat{l}| = \rho \text{ and } \hat{l} \cdot \bar{l} = 0\},$$

where ρ is some constant and has units of length. Note the first three elements $\hat{l}$ correspond to the direction of the line, scaled by ρ and the last three $\bar{l}$ to the moment

of the line. Often in the literature, ρ is chosen to have unit length, here we have chosen to show the effect of scaling throughout this discussion by not making this choice.

2.2 Operations on Lines

Formulas for the angle, minimum distance and moment between two lines are given in [7]. Let $x, y \in L$ be two lines. The angle α between the lines is

$$\alpha = \sin^{-1} \frac{|\hat{x} \times \hat{y}|}{\rho^2} \quad \alpha \in [0, \pi/2]$$

and the minimum distance is

$$d_m = \rho|(x \otimes y)|/|\hat{x} \times \hat{y}| \quad d_m \geq 0,$$

where $x \otimes y$ denotes the moment between two lines defined by

$$x \otimes y = \hat{x} \cdot \bar{y} + \bar{x} \cdot \hat{y}.$$

Note that parallel lines have a moment of zero so that the d_m is also zero for parallel lines.

Let ${}^{j}l$ be a line represented in coordinate frame A_j. Its representation in coordinate frame A_i is given by

$${}^{i}l = {}^{i}_{j}P\,{}^{j}l = \begin{bmatrix} {}^{i}_{j}R & 0 \\ {}^{i}_{j}X\,{}^{i}_{j}R & {}^{i}_{j}R \end{bmatrix} {}^{j}l,$$

where ${}^{i}_{j}R$ is the 3×3 rotation matrix from A_j to A_i and ${}^{i}_{j}X$ is the 3×3 antisymmetric matrix which yields the cross product of the origin of the A_i with respect to A_j with ${}^{i}_{j}R$. The inverse of ${}^{i}_{j}P$ is denoted ${}^{j}_{i}P$ and is equal to the matrix formed by transposing each of the component matrices.

In the application of line geometry to trajectory generation for mechanisms, it is necessary to determine a unique measure of the error in alignment between to lines. This measure must weight both the error in the alignment of the directions and the moments. One that has proven useful in our work is:

$$d(x, y) = \sqrt{|\hat{x} - \hat{y}| + |\bar{x} - \bar{y}|} \tag{1}$$

Note, since ρ is used in the scaling of the directions of the lines, this distance will reflect that scaling. The advantage of this distance over a weighted sum of the angle between and the distance between two lines, such as:

$$\sqrt{d_m^2 + \rho^2\alpha^2}, \tag{2}$$

is that measure(2) is zero for two lines that are parallel but not necessarily collinear. We will use the measure of equation(1) to generate line trajectories for WAM-1 with ρ equal to the length of the third link which provides a balance between errors in angle and errors in distance.

3 Line Kinematics

In order to generate motions for a manipulator in line space, we must determine the forward kinematic relationship. This function will determine the line (or assemblage of lines) determined by a set of joint angles. The derivative of this function, with respect to the joint angles, relates velocities in joint space to line velocities. This same Jacobian can be used, via virtual work, to determine the relationship between joint torques and line forces. This will permit us to define and control line stiffness. Lastly, for a manipulator with four or fewer freedoms we can use the Jacobian to determine by integration the volume of lines reachable by the manipulator.

3.1 Kinematic Transformations

Suppose a mechanism is defined as a serial assemblage of lines connected with joints. The transformation matrices ${}^{i}_{j}P$ will be a function of the intervening joint positions $\Theta = (\theta_1, \ldots \theta_n)$. Therefore we can define a function from joint space, Θ the set of all possible joint angles, to the line space L defined by the last line,

$$f : \Theta \rightarrow L, \tag{3}$$

where

$$f(\theta) = {}^{i}_{j}P(\theta)\ l', \tag{4}$$

where l' is the line coordinates of a line in its own frame or $(\begin{matrix} 0 & 0 & \rho & 0 & 0 & 0 \end{matrix})$. In general the function f is not invertible, since that would require the function to be one-to-one. However if the dimension of Θ is less than or equal to four, f can be finite-to-one, as in the case of the WAM-1. This function can be applied repeatedly to generate trajectories.

3.2 Line Trajectories

A line trajectory is a continuous function from an closed interval to line space. If the image of the interval is in the workspace and the function from joint space to line space is invertible, a trajectory in line space can be mapped into joint space.

A convenient line trajectory, between an initial and final line, is a linear interpolation of distance and angle along the common normal between two lines. Given a length of time t, a recursive bounded deviation joint path can be generated from l_1 to l_2 by using the distance measure of equation(1), [8]. This is an extension of the recursive bounded deviation method for Cartesian interpolation described in [9]. The trajectory is generated by recursively refining a set of intermediate lines lying along the path between l_1 and l_2.

First, the inverse kinematics is solved to compute the joint parameters that will place the link line on l_1 and on l_2. Then, the distance between the desired line at

$t/2$ and the line produced by straight line joint interpolation at $t/2$ is computed. If this distance is less than the maximum deviation the computation is complete. Otherwise, we redefine l_2 to be the desired line at $t/2$ and try the procedure again with $t = t/2$. Tests for singularities and limits on the maximum joint velocities are also necessary.

Using this procedure and the operations on lines so far defined, the motion of a manipulator can be controlled with lines. For some tasks, (eg. pushing and searching) this is more powerful way to think about accomplishing a task than endpoint motions. For example, since stable pushing is most easily accomplished with a two point contact, which can be generated by contact with a line segment, the motion of the segment can be controlled by specifying the motion of the whole line. Of course, in planning the motion the constraint of the limited length of the robot link must also be introduced. This leads to a procedure where first all possible line locations that lead to stable pushing are generated. Then this set is pruned with the constraint of reachability of the line by the manipulator, and then finally the resulting set is pruned with the limited length of the link constraint. Other procedures which exploit the ability to specify the motion of the manipulator with lines can be used to search the reachable space. An example of this is described in section 5.

3.3 Differential Transformation

For differential motions and compliant tasks the concept of a line Jacobian and line stiffness must be introduced. The derivative of f is the *line Jacobian* J, which is simply the $6 \times n$ matrix of partial derivatives of f with respect to θ:

$$J_{ij} = \frac{\partial f_i}{\partial \theta_j}.$$

In general, J will have rank $r \leq \min(4, n)$.

In the line frame, the first three rows of the Jacobian relate joint velocities to the velocity (in [length]/[time]) of the tip of the vector $\rho\hat{l}$. Since motions in the direction of the line, (the third coordinate in this definition) are undefined, the third row of the Jacobian is identically zero. The last three rows of the Jacobian relate joint velocities to the velocity of the tip of the vector $\tilde{l}$. Because of the constraint $\hat{l} \cdot \tilde{l} = 0$, we muse have the differential relationship $d\hat{l} \cdot \tilde{l} + \hat{l} \cdot d\tilde{l} = 0$. Referring to figure 2, a line in its own frame, we see that this relationship yields $[d\tilde{l}]_3 = R[d\hat{l}]_2/\rho$, where the subscripts denote subcomponents of the vectors. This relationship is enforced by the linear dependence of the last row of the Jacobian with the second row.

3.4 Line Stiffness

Complementary to line velocities, we define *line force* to be a set of generalized forces that tend to displace the line along its generalized coordinates; that is, a line

force p on a line l is a sextuple $p = (p_1, \ldots, p_6)$ in the tangent space $P_l = T_l L$ of the line space L evaluated at l. Hence *line stiffness* is the 6×6 matrix K that relates line position to line force,

$$p = Kl.$$

This is a 6×6 rigid body stiffness defined in the usual sense. The projection of the this stiffness from the line space into joint space via

$$\tau = J^T p.$$

reduces the rank of the stiffness to four. Therefore, the resulting joint stiffness K_θ, as related to line stiffness K_l, is given by

$$K_\theta = J^T K_l J.$$

This stiffness matrix will result in a zero stiffness along the direction of the line and for twists about the line. The other stiffness will generate torques to correct errors in the direction of the line and in the moment of the line.

3.5 Line-Workspace

The Jacobian can also be used to find the "volume" of lines that can be reached by a particular manipulator with four or fewer degree-of-freedom. This provides a measure of the line-workspace for a particular manipulator.

Let $W = f(\Theta)$ be the subset of line space defined by the image of Θ under f in equation 3, which we will call the *line workspace*. The boundaries of W in L correspond to singularities of the workspace and are found by examining the rank of J. Singularities occur where the rank of J is less than r. The volume of the workspace, for $n \leq 4$, is

$$v(W) = \int_W dW = \int_\Theta \sqrt{\det(J^T J)} d\Theta. \tag{5}$$

Since J is homogeneous, the volume will have units of $[length]^4$. However, homogeneity was produced by the introduction of the scaling factor ρ, and consequently the measure will depend on this value. In comparing different manipulators a uniform choice of ρ must be used. An example of this workspace is given in section 4.3.

4 Whole-Arm Manipulator Line Kinematics

We have applied these ideas to the WAM-1 constructed at the Massachusetts Institute of Technology's Artificial Intelligence Lab. This manipulator has four freedoms with the Denavit-Hartenberg parameters given by the following table.

coordinate frame number i	link twist (radians) α_{i-1}	link length a_{i-1}	link offset d_i	joint angle θ
1	0	0	0	θ_1
2	$\pi/2$	0	0	θ_2
3	$-\pi/2$	0	L_3	θ_3
4	$\pi/2$	A_3	0	θ_4
5	$-\pi/2$	$-A_4$	L_4	0

Denavit-Hartenberg parameters of the MIT Whole-Arm Manipulator.

The values L_3 and L_4 are the inner and outer link lengths, 22 inches and 17.5 inches and A_3 and A_4 are the offsets in the manipulator, 1.6 inches and 1.1 inches. See figure 1 for a sketch of the kinematics.

In this paper, we will consider an idealized version of the WAM-1 with kinematics offsets A_3 and A_4 set to 0. For the detailed forward and inverse line kinematics of the true manipulator see [8].

4.1 Forward Kinematics

The forward kinematics of a line is a transformation of coordinates from a local to a global reference frame where the transformations depend on the joint angles. The coordinates of the line through the last link of WAM-1, defined with respect the last frame, is

$$l = \begin{bmatrix} 0 & 0 & \rho & 0 & 0 & 0 \end{bmatrix}^T,$$

and, defined with respect to the base frame, is

$$f(\Theta) = {}^0_1P(\theta_1){}^1_2P(\theta_2){}^2_3P(\theta_3){}^3_4P(\theta_4){}^4_5P \;\; l. \tag{6}$$

Each ${}^i_jP(\theta_i)$ in equation 6 is a function of the Denavit-Hartenberg parameters for a single joint. Applying these parameters to each ${}^i_jP(\theta_i)$ and multiplying yields the forward kinematic relation

$$f(\Theta) = \begin{bmatrix} \rho\left((S_1\, S_3 - C_1\, C_2\, C_3)\ S_4 - C_1\, C_4\, S_2\right) \\ \rho\left((-C_1\, S_3 - C_2\, C_3\, S_1)\ S_4 - C_4\, S_1\, S_2\right) \\ \rho(-C_3\, S_2\, S_4 + C_2\, C_4) \\ (C_1\, C_2\, L_3\, S_3 + C_3\, L_3\, S_1)\ S_4 \\ (C_2\, L_3\, S_1\, S_3 - C_1\, C_3\, L_3)\ S_4 \\ L_3\, S_2\, S_3\, S_4 \end{bmatrix}.$$

The set of lines that can be reached by the manipulator is defined as the line workspace W. Ignoring joint limitations, W consists of all lines through all points

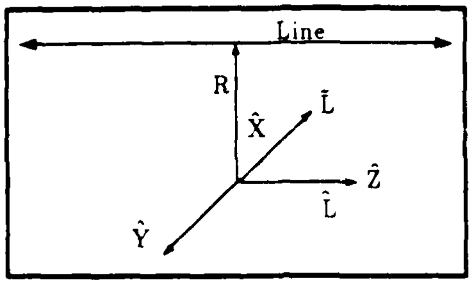

Figure 2: Definition of Line Direction in Line Frame

on a sphere S of radius L_3 traced by the end of the third link. The first two joints position the last joint anywhere on S, while third and fourth joints act as a spherical joint to locate the direction of the line.

The forward line kinematics has two kinds of singularities: the boundaries of W and locations where the axes of the manipulator align with each other or with the line axis decreasing the freedoms available to the robot. The boundary of W consists of all lines that are tangent to S. A manipulator in a configuration that produces one of these lines cannot translate the line further away from the origin. This is the equivalent of the maximum Cartesian reach of a robot. For the WAM-1, this consists of all θ for which $\theta_4 = \pm\pi/2$. The second type of singularity occurs when $\theta_2 = 0$ which aligns the first and third joint axes, and $\theta_4 = 0$, which aligns the third joint axis with the line axis.

These singularities can be found by determining the conditions under which the Jacobian given by equation 7 has rank less than four. The Jacobian J defined with respect to the line frame, see figure 2, is given by

$$J = \begin{bmatrix} -\rho S_2 S_3 & -\rho C_3 & 0 & -\rho \\ -\rho\,(C_2 S_4 + C_3 C_4 S_2) & \rho C_4 S_3 & -\rho S_4 & 0 \\ 0 & 0 & 0 & 0 \\ C_2 C_4 L_3 S_4 - C_3 S_4 L_3 S_2 & S_4^2 L_3 S_3 & C_4 L_3 S_4 & 0 \\ 0 & 0 & 0 & -C_4 L_3 \\ -C_2 S_4^2 L_3 - C_3 C_4 L_3 S_2 S_4 & C_4 L_3 S_3 S_4 & -S_4^2\, L_3 & 0 \end{bmatrix}. \tag{7}$$

The first three rows of J give the change in the direction of the line as a function of the joint velocities and the last three give the change in the moment of the line. The independent motions for $\hat{l}$ in the line frame, are in the $\hat{x}$ and $\hat{y}$ direction. Motions in the direction $\hat{z}$ are not possible as indicated by the zero row of J. The moment of the line may change in length which is equivalent to translating the line in the direction of $\hat{x}$. The moment may also rotate about the direction of the line which is given by the fourth row of J. The last row of J is equal to the second row times $S_4 L_3/\rho$ and ensures that $\hat{l}$ and $\bar{l}$ remain perpendicular when the direction of the line is changed in the $\hat{y}$ direction.

4.2 Inverse Kinematics

The inverse kinematics for a line is solved by a series of geometric transformations. A line that does not pass through the origin of the global coordinate system (which is located at the intersection of the first three axes of the robot) defines a plane with the origin in which the inverse kinematics can be solved. Lines that pass through the origin can be specified with only three parameters, so a constraint on the solution is lost and a single infinity of solutions is generated for our four degree-of-freedom arm. We first examine the general case and then show how the procedure can be applied to the degenerate case.

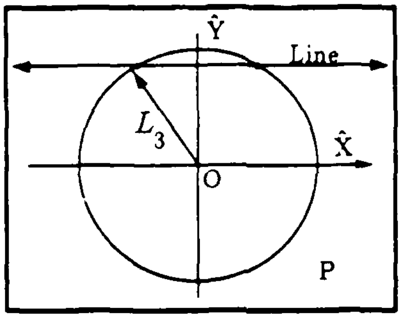

Figure 3: Kinematics in the Plane Defined by the Line and The Origin.

Figure 3 shows a plane defined by a line and the origin O. Points in this frame are related to points in the base frame by a rotation matrix ${}^{P}_{0}R$ formed from the direction and moment of the line. The rotation matrix is chosen such that $\hat{x}$ is in the direction of the line and $\hat{z}$ is perpendicular to the plane P. Then the equation of the line in P, as a locus of points, is simply $y = |\bar{l}|$.

The intersection of the line and the circle of possible locations for joint 4 gives the two possible locations for joint 4 in P. At each of these intersection points, the last link of the robot may be aligned in two directions along the line. In addition, at each location the fourth joint axis can be aligned with either $\pm\hat{z}$ yielding another factor of two. Lastly for each location and each orientation the combination of the first and second joints can be chosen in two different ways. Thus, there are in general sixteen solutions to the inverse kinematics of the simplified WAM-1. (The addition of the offsets increases the number of possible locations of joint 4 to four, however at each location there is only one choice of $\hat{z}$ thus the number of solutions is still sixteen.)

From figure 3, the four possible values θ_4 are $\theta_4 = \pm\sin^{-1}(|\bar{l}|/L_3)$ and $\theta_4 = \pm(\pi - \sin^{-1}(|\bar{l}|/(L_3)))$. The values θ_1 and θ_2 place the fourth joint axis at one of the four possible intersection points

$$ {}^{0}_{P}R \begin{bmatrix} L_3C_4 \\ L_3S_4 \\ 0 \end{bmatrix} = \begin{bmatrix} l \\ m \\ n \end{bmatrix}. $$

We have defined $[\,l \quad m \quad n\,]$ for convenience. This constraint becomes

$$\begin{bmatrix} -C_1S_2L_3 \\ -S_1S_2L_3 \\ C_2L_3 \end{bmatrix} = \begin{bmatrix} l \\ m \\ n \end{bmatrix}.$$

This is solved for θ_1 and θ_2,

$$\theta_2 = \pm\cos^{-1}(n/L_3) \quad \theta_1 = \tan^{-1}(m/l),$$

producing two solutions for each θ_4.

Lastly to solve for θ_3 we require that the fourth joint axis be aligned with either of $\pm\hat{z}$,

$$\begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix} = {}^0_pR \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = {}^0_4R \begin{bmatrix} 0 \\ 0 \\ \pm 1 \end{bmatrix},$$

where $(\,\alpha \quad \beta \quad \gamma\,)$ are defined for convenience.

This constraint specifies the tangent of θ_3 by:

$$\begin{aligned} C_3 &= C_1\beta - S_1\alpha \\ S_3 &= -C_1\alpha - S_1\beta = \gamma/S_2 \\ \theta_3 &= \tan^{-1}(S_3/C_3). \end{aligned}$$

The resulting procedure produces sixteen unique solutions for the case of a line that does not pass through the origin.

For the degenerate case, the plane P is not defined and we must introduce an additional constraint. It becomes possible to place the last link along the line with any value of θ_3. A convenient method is to use the current value of θ_3 and thus define $\hat{z}$ by the direction of the fourth joint axis. Once the plane is defined the non-degenerate procedure can be applied to solve for the joint angles.

4.3 Workspace

The volume of the workspace W is given in equation 5. Its value will scale with the length of the links, the range of angles that the manipulator may reach, and the scaling parameter ρ. The calculation of this parameter is complicated by multiple configurations that reach a single line. If we consider the WAM-1 without joint limits there are, in general, sixteen ways to reach any given line. The volume of the workspace therefore equals

$$\left(\int_\Theta \sqrt{det(J^TJ)}d\Theta\right)/16, \quad \Theta = [0,\ 2\pi)^4.$$

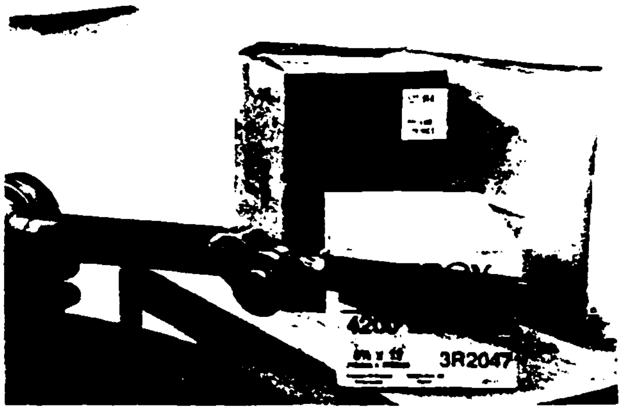

Figure 4: WAM-1 searching over a set of objects.

Figure 5: WAM-1 pushing a box.

In the line frame, the function in this integral is independent of both θ_1 and θ_3. Therefore the integral can be evaluated with respect to θ_4 and θ_2 from 0 to $\pi/2$ and then multiplied by $4\pi^2$. Evaluation in this way eliminates the possibility of multiple covers and the division by sixteen in not necessary. The result is

$$\frac{4\pi^2}{3}\left|\rho^4 - (\rho^3 + L_3^2\,\rho)\,\sqrt{\rho^2 + L_3^2}\right|.$$

Thus, the line workspace has units of the fourth power of length and is dependent on the scaling parameter and the length of the third link. By setting the scaling parameter equal to the length of the second link L_4, the measure can be made to reflect the limited length of the link which is forming the line.

5 Whole-Arm Manipulator Implementations

Simple demonstrations of pushing and searching were performed to examine the usefulness of line trajectories for accomplishing and thinking about tasks. In both tasks, the ability of the WAM-1 to sense contact forces under joint stiffness control from errors in joint positioning was used to determine the onset of contact with the environment and the direction and location of the contact. A discussion of this approach to force sensing can be found in [8] and [10].

In the pushing task, an object is moved along a table by a line. For a line, the location of contact is unimportant, however for a manipulator the object is constrained to lie along the segment doing the pushing. In this demonstration, we placed a box on the table to guarantee that two contacts would occur with the last link. This still left significant latitude in the location and orientation of the box. The manipulator was commanded to follow a line trajectory along a plane so that contact with the box was occurred somewhere along the trajectory. The operation then aligns the box by pushing and deposits it at a final location (see figure 5).

In the searching task, a number of object were placed on a table and the goal was to follow the edges of the objects with the last link of the manipulator (see figure 4). The search procedure was specified by a line congruence, that is a ∞^2 set of desired lines, and a bottom plane in which to begin the search. All lines perpendicular to the vertical line at the origin form the congruence. Then for any point in space, except for points along the vertical line, there is a unique line associated with that point. Given the plane and the congruence, the link moves along the plane until a contact occurs, then the manipulator stops and the contact location and forces are determined. A new line is generated from the congruence that intersects a point a small distance from the contact location, in a direction perpendicular to both the original line and the contact force. The resulting procedure was able to move over a set of three blocks using the sensed forces to generate the motions and identify contacts. Reliability of this procedure suffered from the difficulty of inferring contact locations and forces from joint torques, for the MIT-WAM, as discussed in [8].

6 Conclusion

Line motion is a natural outgrowth of whole-arm manipulation. For the pushing and searching tasks described above, the exact point of contact between the object and the manipulator can vary along the link. Thus for the a links of the WAM, the motion of the links can be controlled with lines.

In this paper, we reviewed basic line geometry and introduced the line Jacobian, workspace and stiffness of a line. We then applied line kinematics to a serial mechanism in which the links were replaced by lines. Finally we demonstrated two tasks,

searching and pushing, which are both analyzed and implemented through the control of line trajectories.

It should be noted that for the WAM, contacts are not frictionless. Thus forces can be exerted along the length of the line. Thus a "line" can really exert five forces, and in order to model the contact we must talk about "lines" with direction and motions along their length. However, the motion of the manipulator can still be controlled with a pure line.

References

[1] J. Kenneth Salisbury. Whole arm manipulation. In *Robotics Research: The Fourth International Symposium*, Santa Cruz, CA, August 1987.

[2] J. K. Salisbury, W. Townsend, B. Eberman, and D. DiPietro. Preliminary design of a whole-arm manipulation system (WAMS). In *Proc. IEEE International Conference on Robotics and Automation*, 1988.

[3] J. K. Salisbury, B. Eberman, M. Levin, and W. Townsend. The design and control of an experimental whole-arm manipulator. In *Robotics Research: The Fourth International Symposium*, 1989.

[4] M. T. Mason and J. K. Salisbury. *Robot Hands and the Mechanics of Manipulation.* MIT Press, Cambridge, Massachusetts, 1985.

[5] E. J. Wilczynski. *Projective Differential Geometry of Curves and Surfaces.* Chelsea Publishing Company, 1901.

[6] K. H. Hunt. *Kinematic Geometry of Mechanisms.* Oxford University Press, Oxford, 1978.

[7] B. E. Shimano. *The Kinematic Design and Force Control of Computer Controlled Manipulators.* PhD thesis, Stanford, 1978.

[8] B. Eberman. Whole-arm manipulation: Kinematics and control. Master's thesis, MIT, Department of Mechanical Engineering, 1989.

[9] R. H. Taylor. Planning and execution of straight line manipulator trajectories. In *IBM Journal of Research and Development Number 23.*, 1979.

[10] B. S. Eberman and J. K. Salisbury. Determination of manipulator contact information from joint torque measurements. In V. Hayward and O. Khatib, editors, *Experimental Robotics I, First International Symposium*, Montréal, Canada, June 1989. Published in: Lecture Notes in Control and Information Sciences, Springer-Verlag, 1990.